

BBN Systems and Technologies

A Division of Bolt Beranek and Newman Inc.

1

AD-A244 222



BBN Report No. 7324

SIMNET CVCC

SOFTWARE DESIGN DOCUMENT

DTIC
ELECTE
JAN 08 1992
S D D

This document has been approved
for public release and sale; its
distribution is unlimited.

92-00300



92 1 6 086

Report No. 7324

SIMNET CVCC

SOFTWARE DESIGN DOCUMENT

June 1991

Prepared by:

BBN Systems and Technologies
Advanced Simulation
10 Moulton Street
Cambridge, MA 02138 USA

Prepared for:

Defense Advanced Research Projects Agency (DARPA)
1400 Wilson Blvd.
Arlington, VA 22209-2308

This research was performed by BBN Systems and Technologies under Contract Nos. MDA973-89-C-0060 and MDA973-90-C-0061 to the Defense Advanced Research Projects Agency (DARPA). The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policy, either expressed or implied, of DARPA, the U.S. Army, or the U.S. Government.

1991 Bolt Beranek and Newman Inc.

Accession For	
NTIS CRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
A-1	

**APPROVED FOR PUBLIC RELEASE
DISTRIBUTION UNLIMITED**

1. Introduction	1
2. Document Overview	2
3. Software Components	3
3.1 Main	3
3.2 Application Overlays.....	3
3.3 Input Devices	3
3.4 Utilities	3
3.5 Communications and Reports	4
3.6 Map	4
3.7 Navigation	4
3.8 Miscellaneous.....	4
4. Detailed Design	5
4.1 Ivis-Main Directory	5
4.1.1 Header: ivis.h	5
4.1.2 File: main.c	5
4.1.3 File: checkpoint.c	5
4.1.4 File: data.c	6
4.1.5 File: option.c	6
4.1.6 File: version.c	6
4.2 Ivis-Menu Directory	6
4.2.1 File: menu.c	7
4.3 Ivis-Pane Directory	8
4.3.1 File: pane_but.c	8
4.3.2 File: pane_menu.c	8
4.3.3 File: pane_stat.c	8
4.4 Ivis-Misc Directory	9
4.4.1 File: covert.c.....	9
4.4.2 File: cursor.c	9
4.4.3 File: dtg.c	10
4.4.4 File: dump_tree.c	10
4.4.5 File: duty.c.....	10
4.4.6 File: error.c	10
4.4.7 File: fb.c	11
4.4.8 File: utm.c	11
4.5 Ivis-Ovr Directory.....	11
4.5.1 File: ovr_drawv.c.....	12
4.5.2 File: ovr_frien.c.....	12
4.5.3 File: ovr_main.c.....	12
4.6 Over-Icn Directory	13

4.6.1	File: icn_arrow.c.....	13
4.6.2	File: icn_creat.c.....	13
4.6.2	File: icn_del.c.....	14
4.6.3	File: icn_draw.c.....	14
4.6.4	File: icn_find.c.....	14
4.6.5	File: icn_hilit.c.....	15
4.6.6	File: icn_ivis.c.....	15
4.6.7	File: icn_misc.c.....	16
4.7	Over-Cm Directory.....	16
4.7.1	File: cm_draw.c.....	16
4.7.2	File: cm_load.c.....	17
4.7.3	File: cm_main.c.....	17
4.8	Cm-Lines Directory.....	17
4.8.1	File: arrows.c.....	18
4.8.2	File: spline.c.....	18
4.9	Cm-Symbols Directory.....	18
4.9.1	File: aviation.c.....	18
4.9.2	File: comm.c.....	19
4.9.3	File: common.c.....	20
4.9.4	File: frills.c.....	21
4.9.5	File: ground.c.....	21
4.9.6	File: group_size.c.....	21
4.9.7	File: gun_types.c.....	23
4.9.8	File: inst.c.....	23
4.9.9	File: letters.c.....	23
4.9.10	File: points1.c.....	24
4.9.11	File: points2.c.....	25
4.9.12	File: role1.c.....	25
4.9.12	File: role2.c.....	25
4.9.13	File: role3.c.....	26
4.9.13	File: role4.c.....	27
4.9.14	File: size.c.....	27
4.9.15	File: Symbols.c.....	28
4.9.16	File: type.c.....	28
4.9.17	File: type1.c.....	28
4.9.18	File: type2.c.....	29
4.9.19	File: type3.c.....	29
4.9.20	File: type4.c.....	29
4.9.21	File: type5.c.....	30
4.9.22	File: type6.c.....	30
4.9.23	File: type7.c.....	31
4.9.24	File: vehicles.c.....	31
4.9.25	File: weapons.c.....	32
4.10	Point Directory.....	33
4.10.1	File: cal-design.c.....	33
4.10.2	File: cal-touch.c.....	33
4.10.3	File: designator.c.....	33
4.10.4	File: dtad.c.....	34
4.10.5	File: touch.c.....	34
4.11	Ivis-Inp Directory.....	34

4.11.1	File: inp_desig.c	34
4.11.2	File: inp_point.c	35
4.11.3	File: inp_touch.c	35
4.12	Ivis-Util Directory	36
4.12.1	File: blast.c	36
4.12.2	File: dimen.c	36
4.12.3	File: listen.c	36
4.12.4	File: send.c	37
4.13	Ivis-Com Directory	37
4.13.1	File: com_action.c	37
4.13.2	File: com_alert.c	38
4.13.3	File: com_net.c	38
4.13.4	File: com_radio.c	39
4.13.5	File: com_recv.c	39
4.13.3	File: com_riu.c	40
4.14	Ivis-Rpt Directory	41
4.14.1	File: rpt_misc.c	41
4.14.2	File: rpt_new.c	41
4.14.3	File: rpt_nmenu.c	42
4.14.4	File: rpt_parse.c	43
4.14.5	File: rpt_print.c	43
4.14.6	File: rpt_show.c	44
4.14.7	File: rpt_smenu.c	44
4.14.8	File: rpt_value.c	45
4.15	Ivis-Fil Directory	45
4.15.1	File: fil_main.c	46
4.15.2	File: fil_rpt.c	46
4.16	Ivis-Map Directory	47
4.16.1	File: map_init.c	47
4.16.2	File: map_main.c	47
4.16.3	map_menu.c	48
4.17	Ivis-Nav Directory	49
4.17.1	File: nav_draw.c	49
4.17.2	File: nav_drive.c	49
4.17.3	File: nav_file.c	49
4.17.4	File: nav_main.c	50
4.17.5	File: nav_menu.c	50
4.17.6	File: nav_msg.c	50
4.18	Bitmaps Directory	51
4.19	Overlays Directory	52
4.20	Resources Directory	52
4.21	Scripts Directory	53
APPENDIX A: DIRECTORY, HEADER, and LIBRARY NAMES		54
APPENDIX B: IVIS SOFTWARE COMPONENT CONFIGURATION		55

1. Introduction

The InterVehicle Information System (IVIS) provides different types of reports that you can create, save, send, receive, delete and relay. Ivis also supports Position Navigation System (POSNAV) which helps you navigate your tank more efficiently and effectively. IVIS is integrated into a larger system know as the Combat Vehicle Command and Control System (CVCC).

2. Document Overview

This document will described the IVIS software components for the system. The eight software components are as follows:

- a. Main
- b. Application Overlays
- c. Input Devices
- d. Utilities
- e. Communications and Reports
- f. Map
- g. Navigation
- h. Miscellaneous

3. Software Components

3.1 Main

Ivis-Main, Ivis-Menu, Ivis-Misc and Ivis-Pane are the four directories associated with the Main software components. Ivis-Main contains the main data handling of command line options, IVIS version number, checkpoint, and recover. All IVIS executables are compiled here. Ivis-Menu contains the code for managing menus and menus stacks. Ivis-Misc provides all the miscellaneous IVIS utility functions. Ivis-Pane provides the code for laying out the IVIS panes for the menu, button, and the status panes.

3.2 Application Overlays

This is the code which drives all of the map application functions. There are five directories which interact with each other (Ivis-Ovr, Over-Icn, Over-Cm, Cm-Lines, Cm-Symbols). Ivis-Ovr contains modules for drawing our own vehicle and other friendly vehicles besides updating vehicle location. It also calls functions in Over-Cm and Over-Icn which are children to Ivis-Ovr.

Over-Cm contains code for loading, drawing and keeping track of control measure overlays. Over-Icn provides the code for creating, displaying, highlighting, deleting, and finding report icons.

The last two directories (Cm-Lines, Cm-Symbols) are logically children of Over-Cm. Cm-Lines contain the code for calculating spline curves and arrows. Cm-Symbols provides the code for drawing NATO symbols.

3.3 Input Devices

This component contains the two directories Ivis-Inp and Point. Ivis-inp is the interface for input devices. Point contains the low level code, non-IVIS specific, for reading from the touch screen and the thumb designator. Point contains executables for calibrating the touch screen and designator.

3.4 Utilities

This contains only one directory called Ivis-Util. This directory contains the code to generate vehicle appearance packets and to broadcast them at a substantial rate. It is also capable of generating and sending certain IVIS reports on a specified network. Lastly, it can listen for and print certain packets of interest to IVIS experimenters.

3.5 Communications and Reports

This component consist of three directories (Ivis-Com, Ivis-Rpt, Ivis-Fil) which is responsible for the handling of communication of reports. Ivis -Com provides the code for communications, instrumentation, handling the receive queue, and sending/receiving reports and routes. Ivis-Rpt consist of the code that generates and shows reports. Ivis-Fil is the code for handling files for saving reports and routes.

3.6 Map

Ivis-Map is the only software directory associated with this component. This is where the code for the IVIS map function are found. Ivis -Map handles all low level map functions such as colors and framebuffers.

3.7 Navigation

This component consist of the Ivis-Nav directory. It provides the code for specifying and displaying navigation routes, and sending packets to the driver's display.

3.8 Miscellaneous

This component includes four directories (Bitmaps, Overlays, Resources, and Scripts). Bitmaps directory are binary data that holds the information for target icons and waypoints. Overlays provides the control measure overlay files. Resources is where all ivis resources are found Scripts are the necessary unix script files for distributing, backing up, calibrating and starting the IVIS system

4. Detailed Design

This section describes the numerous source code files found in their perspective directory written in the C programming language. Each directory comprises individual source files that deals with specific Ivis functions. All header files are affiliated with the include directory which is the library's interface to the world. Private functions that have to be declared globally are either declared in private header files within the include directory or in the source files themselves.

4.1 Ivis-Main Directory

Path: /simnet/ivis/ivis-main

4.1.1 Header: ivis.h

Path: /simnet/ivis/include/ivis.h

This file declares all the constants and data structures that determines IVIS behavior, intervals at which time-based activities are performed in milliseconds, timer durations in ticks, map features, dialog colors, and global data storage. Ivis.h contains some external that are global which can be accessed by other function such as mode for systems communication, pointing devices, display modes, execution modes, and scrolling modes.

4.1.2 File: main.c

Path: /simnet/ivis/ivis-main/main.c

This function contains the Ivis simulator program entry point, initialization code, and main event loop.

FUNCTION NAME	DESCRIPTION
Main	program entry point
CreateDisplay	create widgets defining the layout of ivis display
InitRootWindow	black out root window and disable display
InitProcessTimeout	initialization of process timeout
ProcessTick	call every tick to perform activity

4.1.3 File: checkpoint.c

Path: /simnet/ivis/ivis-main/checkpoint.c

First this functions checks to see if we are in the recovery mode . If yes, recovery is done, and the recovery file is saved. Next, a checkpointing file for the current IVIS session is opened.

FUNCTION NAME	DESCRIPTION
InitIvisCheckpointing	initialization for checkpointing
CheckpointIvis	save state for ivis

4.1.4 File: data.c

Path: /simnet/ivis/ivis-main/data.c

This function describes global data structures used by the IVIS simulator. IvisData ivis is the statement assigned to this variable. The exact data structure itself is defined in the ivis.h header file found in the include directory.

4.1.5 File: option.c

Path : /simnet/ivis/ivis-main/option.c

This file contains routines for processing command line arguments and resource database entries to obtain information about how IVIS is configured.

FUNCTION NAME	DESCRIPTION
ParseOptions	parse command line arguments and retrieve options
LoadResourceDatabase	load the resource database from various files
PrintUsage	print a message about command line arguments

4.1.6 File: version.c

Path : /simnet/ivis/ivis-main/version.c

This file works in conjunction with the Makefile to compile into a program the date, time, and version number at which the program was linked.

4.2 Ivis-Menu Directory

Path: /simnet/ivis/ivis-menu

4.2.1 File: menu.c

Path: /simnet/ivis/ivis-menu/menu.c

Menu.c are the routines that implement the menu pane of the Ivis display. It initializes the menu package and creates the menu widgets associated with the menu page which is described by MenuDefn object. Menu.c also has the routines that implement menu fields whose values are entered by touching the map or using the laser rangefinder.

FUNCTION NAME	DESCRIPTION
InitMenus	initialize the menu package
CreateMenu	create the widgets associated with a menu page, as described by a MenuDefn object
CreateDataField	create a field displaying a data value
ClearMenu	load a menu's field with unspecified values
RefreshMenu	make a menu coincide with the stored values it represents
MenuVisible	called when a menu becomes visible
MenuVisibleCallback	execute the visible callback
MenuInvisible	called when the frontmost menu becomes invisible
FieldSelection	callback invoked when a data entry field is selected
NextField	returns the next menu field of a particular type
HiliteField	highlight a menu's data entry field
UnhiliteField	unhighlight a menu's data entry field
GetFieldValue	fetch a field's value from memory
PutFieldValue	store a field's value in memory
ValidTDBLocation	valid location for data entry field
EnterLocation	called when a location has been entered
CreatePopUpList	creates the widgets for a pop-up list of choices
DisplayPopUpList	display the list of values for a selection field
PopUpListSelection	callback invoked when a popped-up list elements is selected
CreateRBGroup	create a group of radio buttons
RBGCallback	activation callback procedure for radio button group
CreateTextField	create a text field
RecordTextFields	record the contents of a menu's editable text fields
RecordTextField	called when a text field loses input focus
ToggleButtonCallback	value change callback procedure for toggles
IncrementalTouchCount	callback to increment the touch count for the menu
ShowMenu	called to bring one of the menus to the front
HideTopMenu	callback procedure for hiding top menu widget
FrontmostMenu	return the identity of the frontmost menu, or NULL if their no menu is being displayed

4.3 Ivis-Pane Directory

Path: /simnet/ivis/ivis-pane

4.3.1 File: pane_but.c

Path: /simnet/ivis/ivis-pane/pane_but.c

This function defines and creates the button pane. The button pane is a form widget which contains push buttons and separator widgets.

FUNCTION NAME	DESCRIPTION
CreateButtonPane	initialize the button pane

4.3.2 File: pane_menu.c

Path: /simnet/ivis/ivis-pane/pane_menu.c

This function defines menu colors, creates menu colors, the title ivis menu, and creates all menu pane.

FUNCTION NAME	DESCRIPTION
InitDialogColors	initialize the colormap color cells reserved for widget colors
CreateMenuPane	creates all the different menus (which are form widgets) that contain the various pushbuttons; etc.

4.3.3 File: pane_stat.c

Path: /simnet/ivis/ivis-pane/pane_stat.c

This file implements the status pane in the top right corner of the Ivis display. the pane is represented by a form widget called "status". Its contents are separator gadgets, a push button widget, and drawing areas.

FUNCTION NAME	DESCRIPTION
CreateStatusPane	initialize the pane status information
InitStatusDisplay	procedure called to initialize the status display
UpdateStatusDisplay	work procedure called to update the status display
StatusExposeHandler	expose the status window
HiliteReceiveButton	enable or disable highlighting of the receive button
displaySystemMessage	display a message in the space provided by the system
eraseMessage	erase the system message

4.4 Ivis-Misc Directory

Path: /simnet/ivis/ivis-misc

4.4.1 File: covert.c

Path: /simnet/ivis/ivis-misc/covert.c

This file contains resources type converters used by the IVIS simulator. Convert.c converts strings to a pixel value, a color value, a widget, and a map bits.

FUNCTION NAME	DESCRIPTION
RegisterTypeConverters	register our own resource type converters
CvtStringToPixel	convert string to a pixel
CvtStringToWidget	convert a string to a widget, where the string is the name of the widget
CvtStringToXColor	convert a string to an XColor structure
CvtStringToMapFtrbits	convert the string to MAPFTR_BITS

4.4.2 File: cursor.c

Path: /simnet/ivis/ivis-misc/cursor.c

This function contains routines for handling special purpose cursors. There are two types of cursors one is the standard cursor which is the crosshair cursor that appears under the finger and the wait cursor to display when IVIS is processing long request.

FUNCTION NAME	DESCRIPTION
InitCursors	initialization of cursor
DefineCursor	define the type of cursor

4.4.3 File: dtg.c

Path: /simnet/ivis/ivis-misc/dtg.c

The routines in this file converts internal (UNIX) time to the military "date-time group" representation of date and time.

FUNCTION NAME	DESCRIPTION
TimeToDTGString	convert a time to a DTG string

4.4.4 File: dump_tree.c

Path: /simnet/ivis/ivis-misc/dump_tree.c

This file contains the utility which dumps a widget tree. This is not a utility used by ivis but in fact is used by the developer's to look at a widget tree.

FUNCTION NAME	DESCRIPTION
dump_tree_inner	dumps a child
dump_tree	dumps the whole tree from the root

4.4.5 File: duty.c

Path: /simnet/ivis/ivis-misc/duty.c

Duty.c contains the routines for dealing with duty position and string conversion. It converts a duty position description to a string and a string back to a duty position description. Duty position description is battalion commander, company commander, and platoon members.

FUNCTION NAME	DESCRIPTION
DutyPositionToString	convert a duty position description to a string
StringToDutyPosition	convert a string to a duty position description

4.4.6 File: error.c

Path: /simnet/ivis/ivis-misc/error.c

This file contains routines for reporting program and system errors. It records the error messages in a file called ErrorLog. There are four types of errors fatal system error, fatal program error, system warning, and program warning.

FUNCTION NAME	DESCRIPTION
InitErrorHandling	initialize error handling mechanisms
SystemErrorHandler	log a system error in the error log file
XWarningHandler	log a non-fatal error encountered by X Windows
FatalSystemError	called when a system fatal error has occurred
FatalProgramError	called when a program fatal error has occurred
SystemWarning	called when a warning must be issued for system failure
ProgramWarning	called when a warning must be issued for program failure

4.4.7 File: fb.c

Path: /simnet/ivis/ivis-misc/fb.c

Fb.c contains routines for processing resources database entries to obtain information about how the map display is configured.

FUNCTION NAME	DESCRIPTION
InitDualFbs	initialize the visual planes layer of the software, dual frame buffers required
SetupInvisFb	determines which current frame buffer is currently being displayed and copies its contents into the invisible frame buffer
SwapVisAndInvisFb	Makes the currently visible frame buffer invisible and the currently invisible frame buffer visible

4.4.8 File: .utm.c

Path: /simnet/ivis/ivis-misc/utm.c

The routines in this file convert between world coordinates and UTM strings. They are hardwired for the Ft. Knox terrain database.

FUNCTION NAME	DESCRIPTION
UTMString	given a world location, return a UTM string

4.5 Ivis-Ovr Directory

Path: /simnet/ivis/ivis-ovr

4.5.1 File: ovr_drawv.c

Path: /simnet/ivis/ivis-ovr/ovr_drawv.c

This function draws the map icons representing our own vehicle. It has to store the coordinates for the vehicle location, vehicle orientation, turret angle and map location so that it can be erased and redrawn in a new location. The set of functions within this file are initialization of drawing vehicle icons, update appearance information for own vehicle, draw own vehicle icon, redraw own vehicle icon, draw a vehicle's label under its icon, and draw orientation of a Commander's Independent Thermal Viewer (CITV).

FUNCTION NAME	DESCRIPTION
InitVehicleDrawing	initialize drawing of vehicle icons
UpdateOwnVehicleAppearance	update appearance information for own vehicle
DrawOwnVehicleIcon	draw own vehicle icon
RedrawOwnVehicleIcon	redraw own vehicle icon
DrawVehicleIcon	draw a vehicle icon
DrawVehicleLabel	draw a vehicle's label under its icon
DrawCITV	draw orientation of a CITV

4.5.2 File: ovr_frien.c

Path: /simnet/ivis/ivis-ovr/ovr_frien.c

This file tracks and displays other friendly units. Information recorded about friendly vehicles are store in a vehicle table. The set of functions found within this file are icons used for drawing friendly vehicles, initialization of data structures for tracking friendly unit status, update friendly vehicle status, and drawing of icons representing friendly units.

FUNCTION NAME	DESCRIPTION
InitFriendlyUnits	initialize data structures for tracking friendly unit status
UpdateFriendlyVehicleStatus	note the status of the friendly vehicles
DrawFriendlyUnits	draw icons representing friendly units

4.5.3 File: ovr_main.c

Path: /simnet/ivis/ivis-ovr/ovr_main

These are functions for initializing, drawing and checkpointing application features on the map. It calls two functions from ovr_drawv.c to initialize vehicle drawing of icon and to draw own vehicle icon. Ovr_main.c calls two other function from ovr_frien.c to initialize friendly vehicle icons and to draw friendly units. This function also sets colorovermap. Colors are either a full-color display or a monochrome display.

4.6 Over-Icn Directory

Path: /simnet/ivis/over-icn

4.6.1 File: icn_arrow.c

Path: /simnet/ivis/over-icn/icn_arrow

This file calculates and draws arrows for target icons which are off the map. It contains function as follows, draw an arrow to something off the map, given an arrow whose vertices have been calculated, draw it, given a window and a location, calculate the vertices of a triangular arrow which points to the off screen location.

FUNCTION NAME	DESCRIPTION
DrawOffMapArrow	draw an arrow to something off the map
DrawCalceArrow	given an arrow whose vertices have been calculated, draw it
CalcOffMapArrow	given a window and a location, calculate the vertices of a triangular arrow which points to the off-screen location

4.6.2 File: icn_creat.c

Path: /simnet/ivis/over-icn/icn_creat.c

These are functions for creating and deleting icon instances and sets. The function are create an empty set of map icons, create anew instance of a map icon, starting with specified icon discard n of them, and discard an instance of a map icon.

FUNCTION NAME	DESCRIPTION
CreateMapIconSet	create an empty set of map icons
NewMapIconn	create a new instance of a map icon
DiscardNIcons	starting with the specified icon, discard n of ther
DiscardMapIcon	discard an instance of a map icon

4.6.2 File: icn_del.c

Path: /simnet/ivis/over-icn/icn_del.c

This is the menu definition and routines for deleting post to map icons. One function used in this file is to decide which mode we are in and take the appropriate action. If post to map mode, it highlights the icon or if receive mode, it displays appropriate error message. Another function highlights all post to map target icons that have been around longer than the specified time. Lastly, there is a function that deletes user highlighted icons.

FUNCTION NAME	DESCRIPTION
PostToMapIconTouched	if post to map mode, highlight the icon if receive mode, display appropriate message
HilitePostToMapIcons	given time in minutes, highlight all post to map target icons that have been around longer than the specified time
DeletePostToMapIcons	user wants to delete highlighted icons
PostToMapMode	1 if oppFeatureMenu is frontmost, 0 otherwise

4.6.3 File: icn_draw.c

Path: /simnet/ivis/over-icn/icn_draw.c

This file draws target icons and their highlighted boxes. The functions included in this file are get and set constants needed for drawing icons, draw a set of icons given an icon set, draw an icon given a particular icon, draw an icon at that point given (x,y) window coordinates, erase a given icon's highlight, and draw the highlight box.

FUNCTION NAME	DESCRIPTION
InitIconDraw	get and set constants needed for drawing icons
DrawIconSet	given an icon, blinking, and arrow indicators, draw the icon
DrawIconAtPoint	given an icon and an (x,y) window coordinate, draw the icon at that point
EraseHiliteBox	given an icon, erase it's Highlighting
DrawHilteBox	given an icon and highlight style, draw the highlight box

4.6.4 File: icn_find.c

Path: /simnet/ivis/over-icn/icn_find.c

This is functions for finding an icon in a set at given point. The set of function include are find the icon that has last been selected and reset it, search for an icon, and given a point in a map coordinates is the icon there.

FUNCTION NAME	DESCRIPTION
FindAndResetSelectedIconInSet	given a set, find the icon that has last been selected and reset it
FindIconInRotatingSetAtPoint	given a set, beginning at the last selected icon, if no icon found, start again at the beginning of set
FindIconInSetAtPoint	given a set, find icon at specified point
PointOnIcon	given a point in map coordinates , see if it's on icon

4.6.5 File: icn_hilit.c

Path: /simnet/ivis/over-icn/icn_hilit.c

This file implements highlighting of target icons. Icons are highlighted by putting a white square around it. The set of function included in this file are draw a highlighted icon, erase highlighting, and create highlight icons for a report.

FUNCTION NAME	DESCRIPTION
DrawHiliteIcons	draw the highlighted icon set with white box highlighting
DrawHiliteIcon	draw an individual icon with white box highlighting
UnhiliteIcons	erase highlighting and discard highlighted set
UnhiliteIcon	erase and discard individual highlighted icon
HiliteReport	create and draw highlighted icons for a report
HiliteIcon	create and draw a highlight icon

4.6.6 File: icn_ivis.c

Path: /simnet/ivis/over-icn/icn_ivis.c

This file contains functions for creating and displaying of ivis target icons. This is one level of abstraction higher than target icons themselves. This code is for ivis-specific target icons.

FUNCTION NAME	DESCRIPTION
IconsNeedBlinking	indicates if icon should be blinked or not
InitTargetIcons	initialize icon sets, icon bitmaps and icon drawing
CreateReceiveIcons	given a report, create icons for receive set
CreatePostToMapIcons	given a report, make post to map icons
CreateMapIcon	create a report map icon
GetIconSet	given an icon set type, return the set
DrawAllMapIcons	draw all icons displayed on the map
FindMapIconMode	find out if we should be searching for an icon
SelectMapIcon	given an (x,Y) position, select icon
NumberOfIconsAt Point	count number of icons stacked at point
CheckpointPostToMapIcons	first, count the number of icons and store the total; then store each icon in a specified format
RecoverPostToMapIcons	first read the number of icons; then load them into the postToMapIconsSet icon queue

4.6.7 File: icn_misc.c

Path: /simnet/ivis/over-icn/misc.c

This is where all miscellaneous target icon functions are. It also loads the bitmaps used for a particular icon.

FUNCTION NAME	DESCRIPTION
LoadMapIconBitmaps	load the bitmaps used for a particular icon

4.7 Over-Cm Directory

Path: /simnet/ivis/over-cm

4.7.1 File: cm_draw.c

Path: /simnet/ivis/over-cm/cm_draw

This file draws a control measure overlay. It overlays the display, window, component, list, foreground color, background color, friend color, foe color, map window, and map origin.

FUNCTION NAME	DESCRIPTION
DrawOverlay	draws a control measure overlay

4.7.2 File: cm_load.c

Path: /simnet/ivis/over-cm/cm_load.c

This program opens a file containing a control measure overlay and loads it into memory. The file is held in memory by a linked list of OverlayComponent dat structure. The input is of type char (*filename) which is the name of the overlay file. The output is OverlayComponentPtr cmRoot which is the overlay components . A 1 (true) means the file is OK and 0 (false) means the file is not found.

FUNCTION NAME	DESCRIPTION
LoadOverlayFromFile	opens the overlay file and loads it into a linked list of OverlayComponent structures

4.7.3 File: cm_main.c

Path: /simnet/ivis/over-cm/cm_main.c

This file manages all overlays. It selects and remembers current overlays. Also, it keeps list of all overlays previously seen and manages overlay selection menu.

FUNCTION NAME	DESCRIPTION
InitCmOverlay	initialize control measure overlays
InitOverlayList	create the overlay menu; set global variables that reflect widget id's of widgets we will need to modify later
SelectOverlat	select an overlay for display
DrawSelectedOverlay	draw the overlay which is currently selected
AddOverlayToList	add an overlay name to a stored list of names
ShowOverlays	Cause a selected overlay to appear
SelectOverlayChoice	this routine is called as a callback when an item in the selection list is pressed
CheckpointOverlay	write the filename to the checkpoint file
RecoverOverlay	get the overlay filename from the checkpoint file and load it

4.8 Cm-Lines Directory

Path: /simnet/ivis/cm-lines

4.8.1 File: arrows.c

Path: /simnet/ivis/cm-lines/arrow.c

These are the functions for calculating arrow pipe and head from a line described by an array of floating point (x,y) coordinates.

FUNCTION NAME	DESCRIPTION
cm_CalcArrows	calculate spline curves and open head
calcArrowHeadLoc	calculate arrow head location using a perpendicular slope
calcPipe	get the slope, points, and delta for arrow head
calcOpenHead	calculate an arrow head for a pipe
calcArrowHead	calculate an arrow head for a solid line

4.8.2 File: spline.c

Path: /simnet/ivis/cm-lines/spline.c

Functions for calculating spline curves for control measure lines and arrows.

FUNCTION NAME	DESCRIPTION
cm_Generate_slopes	
cm_Draw_cubic_spline	
calculate_lengths	
set_boundary_conditions	This determines the initial boundary conditions, and the N-1st row is determined by the final boundary condition
fill_arrays	
upper_triangularize	
back_solve	
generate_curve	

4.9 Cm-Symbols Directory

Path: /simnet/ivis/cm-symbols

4.9.1 File: aviation.c

Path: /simnet/ivis/cm-symbols/aviation.c

These are the functions for drawing aviation symbols, fixed wing air force, fixed wing army, bomber, fighter, recon air force, recon army, transport air force light, transport air force medium, transport air force heavy, helicopter air force, helicopter army, attack helicopter, transport army light, transport army medium, transport army heavy, rpv air force, and rpv army.

FUNCTION NAME	DESCRIPTION
draw_aviation	draws on the display window background colors and foreground
draw_fixed_wing_army	displays fixed wing army at (x,y) coordinates
draw_bomber	displays bomber at (x,y) coordinates
draw_aviation_arrowhead	displays aviation arrowhead at (x,y) coordinates
draw_fighter	displays fighter at (x,y) coordinates
draw_aviation_recon	displays aviation recon at (x,y) coordinates
draw_recon_air_force	displays recon air force at (x,y) coordinates
draw_recon_army	displays recon army at (x,y) coordinates
draw_transport_aviation	displays transport aviation at (x,y) coordinates
draw_aviation_medium	displays aviation medium at (x,y) coordinates
draw_aviation_heavy	displays aviation heavy at (x,y) coordinates
draw_transport_air_force_light	displays transport air force light at (x,y) coordinates
draw_transport_air_force_medium	displays transport air force medium at (x,y) coordinates
draw_transport_air_force_heavy	displays transport air force heavy at (x,y) coordinates
draw_helicopter_air_force	displays air force helicopter at (x,y) coordinates
draw_fixed_wing_air_force	displays air force fixed wing at (x,y) coordinates
draw_helicopter_army	displays army helicopter at (x,y) coordinates
draw_attack_helicopter	displays attack helicopter at (x,y) coordinates
draw_transport_army_light	displays transport army light at (x,y) coordinates
draw_transport_army_medium	displays transport army medium at (x,y) coordinates
draw_transport_army_heavy	displays transport army heavy at (x,y) coordinates
draw_rpv	displays rpv at (x,y) coordinates
draw_rpv_air_force	displays air force rpv at (x,y) coordinates
draw_rpv_army	displays army rpv at (x,y) coordinates

4.9.2 File: comm.c

Path: /simnet/ivis/cm-symbols/comm.c

This is the code for drawing various military symbols. (air defense radar, ground sensors, artillery locating radar, adp, direction finding, electronic warfare installation, intercept,

telephone center, teleprinter center, radio, radio dummy, target designator, radio relay, communications center, jamming, and unknown)

FUNCTION NAME	DESCRIPTION
draw_radar	displays radar at (x,y) coordinates
draw_air_defense_radar	displays air defense radar at (x,y) coordinates
draw_ground_sensor	displays ground sensors at (x,y) coordinates
draw_artillery_locating_radar	displays artillery locating radar at (x,y) coordinates
draw_adp	displays adp at (x,y) coordinates
draw_direction_finding	displays direction finding at (x,y) coordinates
draw_electronic_warfare_installation	displays electronic warfare installation at (x,y) coordinates
draw_intercept	displays intercept at (x,y) coordinates
draw_telephone_center	displays telephone center at (x,y) coordinates
draw_teleprinter_center	displays teleprinter center at (x,y) coordinates
draw_radio_symbol	displays radio symbol at (x,y) coordinates
draw_radio	displays radio at (x,y) coordinates
draw_radio_dummy	displays radio dummy at (x,y) coordinates
draw_target_designator	displays target designator at (x,y) coordinates
draw_radio_relay	displays radio relay at (x,y) coordinates
draw_communications_center	displays communications center at (x,y) coordinates
draw_jamming	displays jamming at (x,y) coordinates
draw_unknown	displays unknown at (x,y) coordinates

4.9.3 File: common.c

Path: /simnet/ivis/cm-symbols

These are the functions for drawing various military symbols. (nbc decon, maintenance, quatermaster, engineer, topographic)

FUNCTION NAME	DESCRIPTION
draw_nbc_partial	displays a partial nbc at (x,y) coordinates
draw_nbc_decon	displays a nbc decon at (x,y) coordinates
draw_maintenance	displays a maintenance at (x,y) coordinates
draw_quatermaster	displays a quatermaster at (x,y) coordinates
draw_engineer	displays engineer at (x,y) coordinates
draw_topographic	displays topographic at (x,y) coordinates
draw_traffic_control	displays traffic control at (x,y) coordinates

4.9.4 File: frills.c

Path: /simnet/ivis/cm-symbols/frills.c

This code draws various military weapons symbols. (medium gun, heavy gun, weapon arrowhead, rocket launched, missile)

FUNCTION NAME	DESCRIPTION
draw_medium_gun	displays medium gun at (x,y) coordinates
draw_heavy_gun	displays heavy gun at (x,y) coordinates
draw_weapon_arrowhead	displays weapon arrowhead at (x,y) coordinates
draw_rocket_launched	displays rocket launched at (x,y) coordinates
draw_gun_barrel	displays gun barrel at (x,y) coordinates
draw_self_propelled_weapon	displays self propelled weapon at (x,y) coordinates

4.9.5 File: ground.c

Path: /simnet/ivis/cm-symbols/ground.c

These are the functions that draw ground symbols. (radioactive, front line, wire, mine, antitank mine, antipersonnel mine, mine cluster)

FUNCTION NAME	DESCRIPTION
draw_ground	displays ground at (x,y) coordinates
draw_radioactive	displays radioactive at (x,y) coordinates
draw_front_line	displays front line at (x,y) coordinates
draw_wire	displays wire at (x,y) coordinates
draw_wire_45	
draw_mine	displays mine at (x,y) coordinates
draw_antitank_mine	displays antitank mine at (x,y) coordinates
draw_antipersonnel_mine	displays antipersonnel mine at (x,y) coordinates
draw_mine_cluster	displays mine cluster at (x,y) coordinates

4.9.6 File: group_size.c

Path: /simnet/ivis/cm-symbols/group_size.c

This file draws the symbols for different group sizes horizontally and vertically. (squad, section, platoon, company, battalion, regiment, brigade, divisions, corps, army, and army group)

FUNCTION NAME	DESCRIPTION
draw_group_size	displays a group sizes at (x,y) coordinates
draw_squad_h	displays a squad at (x,y) coordinates horizontally
draw_section_h	displays a section at (x,y) coordinates horizontally
draw_platoon_h	displays a platoon at (x,y) coordinates horizontally
draw_company_h	displays a company at (x,y) coordinates horizontally
draw_battalion_h	displays a battalion at (x,y) coordinates horizontally
draw_regiment_h	displays a regiment at (x,y) coordinates horizontally
draw_brigade_h	displays a brigade at (x,y) coordinates horizontally
draw_division_h	displays a division at (x,y) coordinates horizontally
draw_corps_h	displays a corps at (x,y) coordinates horizontally
draw_army_h	displays a army at (x,y) coordinates horizontally
draw_army_group_h	displays a army group at (x,y) coordinates horizontally
draw_squad_v	displays a squad at (x,y) coordinates vertically.
draw_section_v	displays a section at (x,y) coordinates vertically.
draw_platoon_v	displays a platoon at (x,y) coordinates vertically.
draw_company_v	displays a company at (x,y) coordinates vertically.
draw_battalion_v	displays a battalion at (x,y) coordinates vertically.
draw_regiment_v	displays a regiment at (x,y) coordinates vertically.
draw_brigade_v	displays a brigade at (x,y) coordinates vertically.
draw_division_v	displays a division at (x,y) coordinates vertically.
draw_corps_v	displays a corps at (x,y) coordinates vertically.
draw_army_v	displays a army at (x,y) coordinates vertically.
draw_army_group_v	displays a army group at (x,y) coordinates vertically.

4.9.7 File: gun_types.c

Path: /simnet/ivis/cm-symbols/gun_types.c

This module draws various military weapons symbols. (flat trajectory, air defense weapon, flame thrower, high trajectory, gun, surface to surface gun)

FUNCTION NAME	DESCRIPTION
draw_flat_trajectory	displays a flat trajectory at (x,y) coordinates
draw_air_defense_weapon	displays air defense weapons at (x,y) coordinates
draw_flame_thrower	displays flame thrower at (x,y) coordinates
draw_high_trajectory	displays high trajectory at (x,y) coordinates
draw_gun	displays gun at (x,y) coordinates
draw_surface_to_surface_gun	displays surface to surface gun at (x,y) coordinates

4.9.8 File: inst.c

Path: /simnet/ivis/cm-symbols/inst.c

This file calls all the functions which draw installation role indicators. The installation role indicators has various types which are selected in function draw_inst by a case statement.

FUNCTION NAME	DESCRIPTION
draw_inst	displays the installation role indicators at (x,y) coordinates for various types
draw_circle	displays a circle at (x,y) coordinates
draw_triangle	displays a triangle at (x,y) coordinates

4.9.9 File: letters.c

Path: /simnet/ivis/cm-symbols/letters.c

This file draws all the different letters used in military symbology.

FUNCTION NAME	DESCRIPTION
draw_a	displays an a at (x,y) coordinates
draw_c	displays a c at (x,y) coordinates
draw_d	displays a d at (x,y) coordinates
draw_e	displays a e at (x,y) coordinates
draw_f	displays a f at (x,y) coordinates
draw_g	displays a g at (x,y) coordinates
draw_h	displays a h at (x,y) coordinates
draw_i	displays a i at (x,y) coordinates
draw_l	displays a l at (x,y) coordinates
draw_m	displays a m at (x,y) coordinates
draw_n	displays a n at (x,y) coordinates
draw_p	displays a p at (x,y) coordinates
draw_r	displays a r at (x,y) coordinates
draw_s	displays a s at (x,y) coordinates
draw_t	displays a t at (x,y) coordinates
draw_u	displays a u at (x,y) coordinates
draw_v	displays a v at (x,y) coordinates
draw_w	displays a w at (x,y) coordinates

Note: The functions to draw small letters is just a repeat of the the above functions. The functions name is draw_small_(replace with letter) and will display that letter at (x,y) coordinates.

4.9.10 File: points1.c

Path: /simnet/ivis/cm-symbols/points1.c

This file draws the ground point symbology. (point, start point, release point, point of departure, traffic control point, linkup point, passage point, checkpoint, coordinating point, contact point)

FUNCTION NAME	DESCRIPTION
draw_point	
draw_enemy_point	
draw_start_point	
draw_release_point	
draw_point_of_departure	
draw_traffic_control_point	
draw_linkup_point	
draw_passage_point	
draw_checkpoint	
draw_coordinating_point	
draw_contact_point	

4.9.11 File: points2.c

Path: /simnet/ivis/cm-symbols/points2.c

This module draw various ground point symbols. (earthwork, surface shelter, acp, communications checkpoint, rendezvous point, rally point, pop up point, airfield)

FUNCTION NAME	DESCRIPTION
draw_earthwork	
draw_surface_shelter	
draw_underground_shelter	
draw_point_circle	
draw_acp	
draw_communicaions_checkpoint	
draw_rendezvous_point	
draw_rally_point	
draw_point_bowtie	
draw_pop_up_point	
draw_airfield	displays an air field at (x,y) coordinates

4.9.12 File: role1.c

Path: /simnet/ivis/cm-symbols/role1.c

This file draws various installation role indicator symbols. (cannibal, civilian, decon station, maintenance station, prisoners, salvage, stagglers, subsistence, class ii, pol ground)

FUNCTION NAME	DESCRIPTION
draw_cannibal	displays a cannibal at (x,y) coordinates
draw_civilian	displays a civillian at (x,y) coordinates
draw_decon_station	displays a decon station at (x,y) coordinates
draw_maintenance_station	displays a maintenance station (x,y) coordinates
draw_prisoners	displays prisoners at (x,y) coordinates
draw_salvage	displays salvage at (x,y) coordinates
draw_stragglers	displays stagglers at (x,y) coordinates
draw_subsistence	displays subsistence at (x,y) coordinates
draw_clas_ii	displays class_ii at (x,y) coordinates
draw_pol_ground	

4.9.12 File: role2.c

Path: /simnet/ivis/cm-symbols/role2.c

This file draws various installation role indicators. (pol army aviation, pol air force, pol solid, construction, ammunition, air defense ammo, air force ammo, atp, army aviation ammo, artillery ammo)

FUNCTION NAME	DESCRIPTION
draw_bow_tie	displays a bow tie at *x,y) coordinates
draw_pol_army_aviation	displays pol army aviation at (x,y) coordinates
draw_lemiscate	
draw_pol_air_force	displays pol air force at (x,y) coordinates
draw_pol_solid	
draw_construction	displays construction at (x,xy) coordinates
draw_ammunition	displays ammunition at (x,y) coordinates
draw_air_defense_ammo	displays air defense ammo at (x,y) coordinates
draw_air_force_ammo	displays air force ammo at (x,y) coordinates
draw_atp	
draw_army_aviation_ammo	displays army aviation ammo at (x,xy) coordinates
draw_artillery_ammo	displays artillery ammo at (x,y) coordinates

4.9.13 File: role3.c

Path: /simnet/ivis/cm-symbols/role3.c

This module draws various installation role indicator symbols. (chemical ammo, mines, explosives, nuclear ammo, rocket artillery ammo, small arms ammo, tank ammo, personal demand, major end items, medical material)

FUNCTION NAME	DESCRIPTION
draw_chemical_ammo	displays chemical ammo at (x,y) coordinates
draw_mines_and_explosive	displays mines and explosive at (x,y) coordinates
draw_nuclear_ammo	displays nuclear ammo at (x,y) coordinates
draw_rocket_artillery_ammo	displays rocket artillery ammo at (x,y) coordinates
draw_artillery_ammo	displays artillery ammo at (x,y) coordinates
draw_small_arms_ammo	displays small arms ammo at (x,y) coordinates
draw_tank_ammo	displays tank ammo at (x,y) coordinates
draw_personal_demand	displays personal demand at (x,y) coordinates
draw_major_end_items	displays major end items at (x,y) coordinates
draw_medical_material	display medical material at (x,y) coordinates

4.9.13 File: role4.c

Path: /simnet/ivis/cm-symbols/role4.c

This file draws various installation role indicator symbols. (repair parts, class x, all classes, grave registration, hospital, material management center, parking, topographic installation, water)

FUNCTION NAME	DESCRIPTION
draw_repair_parts	displays repair parts at (x,y) coordinates
draw_class_x	
draw_all_classes	
draw_grave_registration	displays a grave registration at (x,y) coordinates
draw_hospital	displays a hospital at (x,y) coordinates
draw_material_management_center	displays a material management center at (x,y) coordinates
draw_parking	displays parking at (x,y) coordinates
draw_topographic	

4.9.14 File: size.c

Path: /simnet/ivis/cm-symbols/size.c

This module draws all the different unit sizes. (squad, section, platoon, company, battalion, regiment, brigade, division, corps, army, army group, task force, team)

FUNCTION NAME	DESCRIPTION
draw_size	displays the unit size at (x,y) coordinates
draw_squad	displays a squad at (x,y) coordinates
draw_section	displays a section at (x,y) coordinates
draw_platoon	displays a platoon at (x,y) coordinates
draw_company	displays a company at (x,y) coordinates
draw_battalion	displays a battalion at (x,y) coordinates
draw_regiment	displays a regiment at (x,y) coordinates
draw_brigade	displays a brigade at (x,y) coordinates
draw_division	displays a division at (x,y) coordinates
draw_corps	displays a corps at (x,y) coordinates
draw_army	displays army at (x,y) coordinates
draw_army_group	displays army group at (x,y) coordinates
draw_task_force	displays a task force at (x,y) coordinates
draw_team	displays a team at (x,y) coordinates

4.9.15 File: Symbols.c

Path: /simnet/ivis/cm-symbols/symbols.c

This file contains all the driving function for drawing military symbols. It is the main entry point for the cm-symbols directory. The module decides which type of symbol needs to be displayed and goes to the appropriate file in the cm-symbols directory to process it.

FUNCTION NAME	DESCRIPTION
cm_Draw_symbol	processes the type to be displayed

4.9.16 File: type.c

Path: /simnet/ivis/cm-symbols/type.c

This module contains the main driving function for drawing unit symbology. It selects which unit type needs to be displayed

FUNCTION NAME	DESCRIPTION
draw_unit	selects unit type and displays that type at (x,y) coordinates
draw_enemy_unit	displays enemy unit at (x,y) coordinates
draw_box	
draw_partial_box	
draw_subbox	
draw_rotating_box	

4.9.17 File: type1.c

Path: /simnet/ivis/cm-symbols/type1.c

This file draws the various unit military symbols. (infantry, cavalry, armor, mech infantry, armored cavalry, artillery, medical, dental, motorized)

FUNCTION NAME	DESCRIPTION
draw_infantry	displays a infantry at (x,y) coordinates
draw_cavalry	displays a cavalry at (x,y) coordinates
draw_armor	displays armor at (x,y) coordinates
draw_mech_infantry	displays mech infantry at (x,y) coordinates
draw_armored_cavalry	displays armored cavalry at (x,y) coordinates
draw_artillery	displays artillery at (x,y) coordinates
draw_medical	displays medical at (x,y) coordinates
draw_dental	displays dental at (x,y) coordinates
draw_motorized	displays motorized at (x,y) coordinates

4.9.18 File: type2.c

Path: /simnet/ivis/cm-symbols/type2.c

This module draws various military unit symbols. (antiarmor, fixed wing, rotary wing, helicopter, air cavalry, bridging, motorized infantry, bifv dismounted, bifv_mouted)

FUNCTION NAME	DESCRIPTION
draw_antarmor	displays antiarmor at (x,y) coordinates
draw_fixed_wing	displays fixed wing at (x,y) coordinates
draw_rotary_wing	displays rotary wing at (x,y) coordinates
draw_helicopter	displays helicopter at (x,y) coordinates
draw_air_cavalry	displays air cavalry at (x,y) coordinates
draw_bridging	displays bridging at (x,y) coordinates
draw_motorized_infantry	displays motorized infantry at (x,y) coordinates
draw_bifv_dismounted	displays bifv dismounted at (x,y) coordinates
draw_bifv_mounted	displays bifv mounted at (x,y) coordinates

4.9.19 File: type3.c

Path: /simnet/ivis/cm-symbols/type3.c

This file draws various military unit symbols. (supply, supp maint, mountain, signal, ordnance, petrol supply, psychological, rocket artillery, finance)

FUNCTION NAME	DESCRIPTION
draw_supply	displays supply at (x,y) coordinates
draw_supp_maint	displays supp maint at (x,y) coordinates
draw_mountain	displays a mountain at (x,y) coordinates
draw_signal	displays a signal at (x,y) coordinates
draw_ordnance	displays ordnance at (x,y) coordinates
draw_petrol_supply	displays petrol supplt at (x,y) coordinates
draw_psychological	displays psychological at (x,y) coordinates
draw_rocket_artillery	displays rocket artillery at (x,y) coordinates
draw_finance	displays finance at (x,y) coordinates

4.9.20 File: type4.c

Path: /simnet/ivis/cm-symbols/type4.c

This file draws various military unit symbols. (sound ranging, supp trans, surface surface, survey, transportation, unmanned air recon, veterinary, aerial observation, air assault insufficient)

FUNCTION NAME	DESCRIPTION
draw_sound_ranging	displays sound ranging at (x,y) coordinates
draw_supp_trans	displays supp trans at (x,y) coordinates
draw_surface_surface	displays surface surface at (x,y) coordinates
draw_survey	displays survey at (x,y) coordinates
draw_transportation	displays transportation at (x,y) coordinates
draw_unmanned_air_recon	displays unmanned air recon at (x,y) coordinates
draw_veterinary	displays veterinary at (x,y) coordinates
draw_aerial_observation	displays aerial observation at (x,y) coordinates
draw_air_assault_insufficient	display air assault insufficient at (x,y) coordinates

4.9.21 File: type5.c

Path: /simnet/ivis/cm-symbols/type5.c

This file draws various military unit symbols. (air assault sufficient, air force surveillance, nato air borne, nato air assault sufficient, airborne, air defense, amphibious, amphibious engineer, nbc, nbc recon)

FUNCTION NAME	DESCRIPTION
draw_air_assault_sufficient	displays air assault sufficient at (x,y) coordinates
draw_air_force_surveillance	displays air force surveillance at (x,y) coordinates
draw_nato_airborne	displays nato airborne at (x,y) coordinates
draw_nato_assault_sufficient	displays nato air assault sufficient at (x,y) coordinates
draw_airborne	displays airborne at (x,y) coordinates
draw_air_defense	display air defense at (x,y) coordinates
draw_air_defense	displays air defense at (x,y) coordinates
draw_amphibious	displays amphibious at (x,y) coordinates
draw_amphibious_engineer	diplays amphibious engineer at (x,y) coordinates
draw_nbc	
draw_nbc_recon	

4.9.22 File: type6.c

Path: /simnet/ivis/cm-symbols/type6.c

This file draw various military symbols. (adjutant general, ranger, replacement, service, special forces, support, civil affairs, data processing unit, electronic warfare, labor resources)

FUNCTION NAME	DESCRIPTION
draw_adjutant_general	displays adjutant general at (x,y) coordinates
draw_ranger	displays ranger at (x,y) coordinates
draw_replacement	displays a replacement at (x,y) coordinates
draw_service	displays service at (x,y) coordinates
draw_special_forces	displays special forces at (x,y) coordinates
draw_support	displays support at (x,y) coordinates
draw_civil_affairs	displays civil affairs at (x,y) coordinates
draw_data_processing_unit	displays data processing unit at (x,y) coordinates
draw_electronic_warfare	displays electronic warfare at (x,y) coordinates
draw_labor_resources	displays labor resources at (x,y) coordinates

4.9.23 File: type7.c

Path: /simnet/ivis/cm-symbols/type7.c

This file draws various military unit symbols. (meteorological, military police, military intelligence, light infantry, cewi, surface to air, nbc smoke)

FUNCTION NAME	DESCRIPTION
draw_meteorological	displays meteorological at (x,y) coordinates
draw_military_police	displays military police at (x,y) coordinates
draw_military_intelligence	displays military intelligence at (x,y) coordinates
draw_light_infantry	displays light infantry at (x,y) coordinates
draw_cewi	
draw_surface_to_air	displays surface to air at (x,y) coordinates
draw_nbc_smoke	

4.9.24 File: vehicles.c

Path: /simnet/ivis/cm-symbols/vehicles.c

This module draws all military vehicles. Amphibious vehicle, apc, light tank, medium tank, heavy tank, armored engineer vehicle, avlb, bifv, cfv, hovercraft, wheeled cross country engineer tractor, tracked engineer tractor)

FUNCTION NAME	DESCRIPTION
draw_rotating_vehicle	displays a rotating vehicle type at (x,y) coordinates and a angle
draw_vehicle	displays a vehicle type at (x,y) coordinates and a angle
draw_amphibious_vehicle	displays amphibious vehicle at (x,y) coordinate and a angle
draw_apc	
draw_light_tank	displays light tank at (x,y) coordinates and a angle
draw_medium_tank	displays medium tank at (x,y) coordinates and a angle
draw_heavy_tank	displays heavy tank at (x,y) coordinates and a angle
draw_armored_engineer_vehicle	displays armored engineer vehicle at (x,y) coordinates and a angle
draw_avlb	
draw_bifv	
draw_cfv	
draw_hovercraft	displays hovercraft at (x,y) coordinates and a angle
draw_engineer_tractor	displays engineer tractor at (x,y) coordinates and a angle
draw_wheeled_cross_country	displays wheeled cross country at (x,y) coordinates and angle
draw_wheeled_cross_country_engineer_tractor	displays wheeled cross country engineer tractor at (x,y) coordinates and a angle
draw_tracked	displays tracked at (x,y) coordinates and angle
draw_tracked_engineer_tractor	displays tracked engineer tractor at (x,y) coordinates and a angle

4.9.25 File: weapons.c

Path: /simnet/ivis/cm-symbols/weapons.c

This file calls military weapons symbols drawing functions and draws gun top, sides, and bottoms.

FUNCTION NAME	DESCRIPTION
draw_gun_type	displays gun type at (x,y) coordinates
draw_gun_size	displays gun size at (x,y) coordinates
draw_gun_top	displays gun top at (x,y) coordinates
draw_gun_side	displays gun side at (x,y) coordinates
draw_gun_bottom	displays gun bottom at (x,y) coordinates
draw_weapon	displays weapon type at (x,y) coordinates

4.10 Point Directory

Path: /simnet/ivis/point

4.10.1 File: cal-design.c

Path: /simnet/ivis/point/cal-design.c

This file is a stand alone program to calibrate the thumb designator on the Commander's Handle. It initializes the Data Translation Analog to Digital card. Before it enters the calibration routine, it clears the screen and displays the program title. Next it will prompt you to push the thumb designator on the command handle to the limit (left, right, up, and down) but ask you first to determine the center position. For each of the five position , it determines a value and stores it in /simnet.li file.

4.10.2 File: cal-touch.c

Path: /simnet/ivis/point/cal-touch.c

This module contains the program for calibrating the touch screen. It initializes the toolkit for the touch options and retrieve configurations options from the resource database. It opens and initializes the serial port for the touch screen. The program creates a window with touch targets and prompt string. The touch screen is polled every 50 ms.

FUNCTION NAME	DESCRIPTION
CheckTouchScreen	checks touch screen for current action
RecordCalibration	compute and write the results of the calibration

4.10.3 File: designator.c

Path: /simnet/ivis/point/designator.c

This file provides the interface to the thumb designator as a pointing device.

FUNCTION NAME	DESCRIPTION
GetDesignatorResources	retrieve resources describing designator

4.10.4 File: dtad.c

Path: /simnet/ivis/point/dtad.c

4.10.5 File: touch.c

Path: /simnet/ivis/point/touch.c

This module contains the code that provides an interface to a touch screen. It maintains a circular queue of touch positions.

FUNCTION NAME	DESCRIPTION
ReintTouchPositionQueue	reinitialize the circular queue
InsertTouchPosition	inserts a new touch position into circular queue
AverageTouchPosition	takes the average of the touch positions in the circular queue
GetTouchScreenResources	retrieve resource describing touch screen
OpenTouchScreen	open and initialize serial port to the touch screen
ReadTouchScreenCalibration	read touch screen calibration parameters
PollTouchScreen	returns the current state of the touch screen; if the screen is being touched, or has been touched within the last touchTimeout milliseconds, it returns 1 along with the position most recently touched. Otherwise, it returns 0
TouchToDisplay	transform touch screen coordinates to display coordinates; use averaging to avoid sudden jumps when taking finger off the screen

4.11 Ivis-Inp Directory

Path: /simnet/ivis/ivis-inp

4.11.1 File: inp_desig.c

Path: /simnet/ivis/ivis-inp/inp_desig.c

This file provides an interface to the thumb designator on the commander's control handle.

FUNCTION NAME	DESCRIPTION
InitDesignator	initialize the thumb designator as the pointing device and moves the the pointer to an obscure corner
CheckDesignator	called periodically to run the Designator controlled pointer; Cursor is always present on screen; Sends button events as the thumb button is pressed and released; Returns 1 if the designator is currently being used by the user to make inputs to the application; otherwise it returns 0
input_transfer_function	maps a raw input in ADU's (A to D units) to a pointer movement rate in pixel/sec; the transfer function has a deadzone .dead_zone ADU's on either side of .center; the maximum input (from calibration) is .max_input, for which a rate of .speed pixel/sec is returned; Warning: this function depends on the static Scale_Adu_To_Pps calculated in InitDesignator

4.11.2 File: inp_point.c

Path: /simnet/ivis/ivis-inp/inp_point.c

This module tracks all the movement of the pointer across the window.

FUNCTION NAME	DESCRIPTION
InitPointerPosition	initialization of pointer position relative to the root (x,y) coordinates and window (x,y) coordinates
UpdatePointerPosition	compares the initial pointer position with present
DeltaPointerPosition	calculates the slope of the pointer position
CheckMouse	returns 1 if the mouse is currently being used to make inputs to the application; otherwise it is 0; also records the current position of pointer
GrabPointer	confine the pointer to the specified window
UngrabPointer	release the pointer from the last grab

4.11.3 File: inp_touch.c

Path: /simnet/ivis/ivis-inp/inp_touch.c

The routines in this file provide an interface to the touch screen. It checks to see whether the finger is currently down on the touch screen and the last position to which the pointer was wrapped.

FUNCTION NAME	DESCRIPTION
InitTouchScreen	initialize use of the a touch screen as the pointing device
CheckTouchScreen	checks current interaction with touch screen; return 1 if the touch screen is currently being used to make inputs to the application; otherwise it returns 0
SendButtonEvent	send a button press event to the window under the pointer; documentation notwithstanding, events do NOT propagate to parent windows if the window to which the event is sent is not selecting on the event; so we have to try to do it ourselves; this version propagates at most one level (by recording parentages on the way down); this suffices for scrolling buttons

4.12 Ivis-Util Directory

Path: /simnet/ivis/ivis-util

4.12.1 File: blast.c

Path: /simnet/ivis/ivis-util/blast.c

This program is used for testing the IVIS system. It generates vehicle appearance PDU's at a large rate. The file initializes tick duration, prepares a vehicle appearance PDU, and repeatedly broadcast the PDU.

4.12.2 File: dimen.c

Path: /simnet/ivis/ivis-util/dimen.c

This module computes the dimensions in pixels for a window. It takes a 9 inch window with a 4:5 aspect ratio and computes the width and height of the window in pixels per inch. The formulas are as follows:

$$\text{Width} = \text{sqrt} (\text{sqr} (\text{window size}) / (1.0 + \text{sqr} (\text{aspect ratio})))$$

$$\text{Height} = \text{Width} * \text{aspect ratio}$$

$$\text{Pixels} = (\text{Width} * \text{pixel per inch}) * (\text{Height} * \text{pixel per inch})$$

4.12.3 File: listen.c

Path: /simnet/ivis/ivis-util/listen.c

This file listens for, and prints, certain PDU's of interest to IVIS experimenters. It also defines the data structure for associating vehicles ID's with vehicle role. The module checks to make sure there is an exercise id, initializes table vehicle information and initializes the network interface.

FUNCTION NAME	DESCRIPTION
LookupVehicle	locate any entry we have associating a company and bumper number with a vehicle's exercise ID and vehicle ID.
PrintProlog	print the current time and the exercise ID of a noteworthy PDU
PrintVehicle	print the identity of a vehicle known by exercise ID and vehicle ID.

4.12.4 File: send.c

Path: /simnet/ivis/ivis-util/send.c

This program generates and sends certain IVIS reports on specified networks.

Usage:

```
SEND <exercise> <network[<file>]  
  <exercise> identifies the exercise into which the report is to be sent  
  <network> identifies the network on which the report is to be sent  
  <file> is the name of a file describing the report
```

If the report filename is omitted, then details of the report are read from the standard input with prompts.

4.13 Ivis-Com Directory

Path: /simnet/ivis/ivis-com

4.13.1 File: com_action.c

Path : /simnet/ivis/ivis.com/com_action.c

This file implements the action menu used to control processing and routing of a message.

FUNCTION NAME	DESCRIPTION
CreateActionMenu	create an instance of the action menu
ConfigureActionMenu	adjust an Action menu's toggle buttons to describe the networks we are actually tuned to
ShowFreshActionMenu	re-initialize and display an Action menu
ShowActionMenu	bring an Action Menu to the front
SendFromActionMenu	transmit a message on radio networks chosen from an Action menu
SelectHigherNetworks	select networks for sending a message up the chain of command
SelectLowerNetworks	select networks for sending a message down the chain

4.13.2 File: com_alert.c

Path: /simnet/ivis/ivis-com/com_alert.c

The code in this file alerts the operator with audible and visual alerts.

FUNCTION NAME	DESCRIPTION
InitOperatorAlert	initialize state of operator alert module
AudibleAlert	generate an audible alert
PollOperatorAlert	called periodically to make beeps and flash things

4.13.3 File: com_net.c

Path: /simnet/ivis/ivis-com/com_net.c

The routines in this file manage the interface between the IVIS simulator and the simulation Ethernet. It also puts instrumentation packets on the net.

FUNCTION NAME	DESCRIPTION
InitSimNetwork	initialize processing of Ethernet communications
SetupPDUHeader	fill in fields of PDU header
SendIVISBroadcastPDU	send a PDU to other IVIS systems
SendIVISReportGenerationPDU	inputs reportAction, reportType, totalTouches, activeTime, and elapsedTime
SendIVISMessageRetrievalPDU	inputs IVIS_SystemIdentifier originator, messageType, reportType, elapsedTime, actionOnMessage, and retrievalMethod
SendIVISMessageReceiptPDU	inputs messageType and reportType
SendIVISMapUsagePDU	send state of the map
SendIVISItemQuantityPDU	send counts of high priority message in queue, low priority message in queue, target icons, receive icons, screen touches, thumb designator selections, map locations inputs, and laser range finder inputs
SendAlertOperator	send a PDU to invoke a beep in the headset
SendPDU	send a PDU via the network, checking for any error
ReadPDU	read and process PDUs received from the simulation network
GetNewMessageSerialNumber	issues new serial numbers for messages

4.13.4 File: com_radio.c

Path: /simnet/ivis/ivis-com/com_radio.c

This file implements the IVIS system's interface with combat radio networks.

FUNCTION NAME	DESCRIPTION
InitRadioNetworks	initialize information describing radio networks.
RadioNetworkAccessible	returns a 1 if and only if a radio network is accessible
SendRadioMessage	transmit a message on a particular radio network
ReceiveRadioMessage	process message received from another IVIS system

4.13.5 File: com_recv.c

Path: /simnet/ivis/ivis-com/com_recv.c

This file implements the queue of received message. The queue is represented as a doubly-linked list of RcvQueueElement structures.

FUNCTION NAME	DESCRIPTION
InitReceiveQueue	initialize the queue of received messages
EnqueueReceiveMessage	queue up a message that has been received from another Ivis system
SelectQueueMessage	note the selection of a queue message
ReceiveIconTouched	if ReceiveMode, select the message, otherwise display appropriate message
ReceiveMode	is 1 if receiveMenu is frontmost, 0 otherwise
HideMessage	menu is hidden and removes highlight
ShowSelectedMessage	displays a message selected from the queue
DeleteSelectedMessage	delete a message selected from the queue
SaveReportMessage	given a message that is a report, save it
RemoveMessage	removes a message from the queue and the list, and remove it's icons from the queue
ResetInspected	message is no longer being inspected, reset its flag
NumberOfQueueMessages	count the queued messages of a particular priority
CheckpointMessageQueue	this inputs the file descriptor of a checkpoint file.
RecoverMessageQueue	this inputs the file descriptor of the checkpoint file and the amount of time the system was down.
RemoveOldMessageFromQueue	If not in the receive mode check for message which have been in the queue longer than queueWaitTime and are not being inspected

4.13.3 File: com_riu.c

Path: /simnet/ivis/ivis-con/com_riu.c

This file implements the IVIS system's interface with its RIU. Since the RIU will allow only one message at a time for transmission, messages to be transmitted are queue.

FUNCTION NAME	DESCRIPTION
InitRIUInterface	initialize communication with the RIU
PollRIU	give up on the RIU simulator if it doesn't seem to be there
RIUShutdown	If the RIU times out during polling, or the association layer doesn't get a response from the RIU, shut down RIU operations
EnqueueRIUMessage	queue up a message for transmission to our RIU
DequeueRIUMessage	note that a message has been processed by the RIU and hand it the next one
SendNextRIUMessage	send the next message to our RIU for broadcast

4.14 Ivis-Rpt Directory

Path: /simnet/ivis/ivis-rpt

4.14.1 File: rpt_misc.c

Path: /simnet/ivis/ivis-rpt/rpt_misc.c

This file contains miscellaneous routines related to the creation and display of IVIS reports.

FUNCTION NAME	DESCRIPTION
FetchCriticalShortage	summarize the critical shortage fields of a situation report as a string
FetchTime	return the contents of a time field as a string
FetchIntellItem1	return the first line of an intelligence item summary
FetchIntellItem2	return the second line of an intelligence item summary
FetchObsItem	return the second line of an obstacle summary
CreateReportIcons	given a report, create target icons for the report
ReportFileSystemMessage	posttomap icon chosen; give system message that tells which report file to look in
CreateReport	called when the new button is pressed on the reports menu
ShowReportFile	called when the old button is pressed on the reports menu
HideReportMenu	called when the cancel button is pressed on the reports menu

4.14.2 File: rpt_new.c

Path: /simnet/ivis/ivis-rpt/rpt_new.c

This file implements creation of new reports and describes the set of menus for creating a report.

FUNCTION NAME	DESCRIPTION
CreateNewReportMenus	creates menus for creating reports
ShowReportCreationMenu	put up the menu for creating a particular type of report
PreReportCreationMenu	displays the previous menu for creating a particular type of report
NextReportCreationMenu	display the next menu for creating a particular type of report
PostProcessReport	called when the 1st send button is pressed following report creation
SendCreateReport	called when the 2nd send button is pressed following report creation
CancelReportCreation	called when the cancel button is pressed during a report creation
ReviseField	called when a field is selected from a report creation summary page
CreatReportUpdateIcons	update map icons displayed for report
CreateReportVisible	called when report creation menu becomes invisible
PostCreateReport	post a newly created report to the map
OverlayEntered	called back when overlay name has ben entered into FRAGO report
SenReportStats	gather report creation statistics and send them

4.14.3 File: rpt_nmenu.c

Path: /simnet/ivis/ivis-rpt/rpt_newmenu.c

This file contains the definition of menus for creating new reports. The data structure defined in this module is as follows:

DATA STRUCTURE	DESCRIPTION
MenuDefn	list of report creation menus
unsigned char numberMenus	number of report creation menus
unsigned char priority	priority of report
short size	size of report variant
unsigned char postable	1: report can be posted to map
IVIS_Report report	storage for report fields
unsigned char active	1: report currently being created
unsigned char activeMenu	which menu page is currently displayed
ActionInfoPtr actionMenu	menu for specifying report routing

Here is a list of all the reports that are generated:

- Fire Report
- Ammunition Status Report
- Contact Report
- FRAGO Report
- Intelligence Report
- NBC Report
- Shell Report
- Situation Report
- Spot Report

4.14.4 File: rpt_parse.c

Path: /simnet/ivis/ivis-rpt/rpt_parse.c

This file contains a routine for parsing a description for the contents of an IVIS report. It also contains hardwire UTM mapping information for the Ft. Knox terrain database.

FUNCTION NAME	DESCRIPTION
ParseReport	parse a description of a report
ReadField	reads a field value from a file or standard input
StringsEqual	returns 1 if two strings are equal modulo capitalization

4.14.5 File: rpt_print.c

Path: /simnet/ivis/ivis-rpt/rpt_print.c

This file contains code for printing the contents of Ivis datagrams. It also contains hardwire UTM mapping information for the Ft. Knox terrain database.

FUNCTION NAME	DESCRIPTION
PrintDatagram	print the datagram transmitted by an IVIS
PrintReport	print a report
PrintActivity	
PrintAdjust	
PrintDisposition	
PrintHeading	
PrintIntel	
PrintLevel	
PrintLocation	
PrintNumber	
PrintSuplly	
PrintTime	
PrintWhat	

4.14.6 File: rpt_show.c

Path: /simnet/ivis/ivis-rpt/rpt_show.c

This module implement menus for displaying IVIS reports. It also describes the set of menus for displaying a report.

FUNCTION NAME	DESCRIPTION
CreateShowReportMenus	create menus for displaying received reports
ShowReport	display a report
ShowReportAction	Called when the Action button is pressed during report viewing
SendViewReport	called when the send button is pressed
DeleteReport	called when the delete button is pressed
CancelReportViewing	called when a report is no longer being viewed
PostViewReport	post a viewed report to a map
GetReportSize	given a report type, return the size of the report

4.14.7 File: rpt_smenu.c

Path: /simnet/ivis/ivis-rpt/rpt_smenu.c

This file contains the definition of menus for creating existing reports. The data structure defined in this module is as follows:

DATA STRUCTURE	DESCRIPTION
MenuDefn	list of report creation menus
unsigned char numberMenus	number of report creation menus
unsigned char priority	priority of report
short size	size of report variant
unsigned char postable	1: report can be posted to map
IVIS_Report report	storage for report fields
unsigned char active	1: report currently being created
unsigned char activeMenu	which menu page is currently displayed
ActionInfoPtr actionMenu	menu for specifying report routing

Here is a list of all the reports that are generated:

- Fire Report
- Ammunition Status Report
- Contact Report
- FRAGO Report
- Intelligence Report
- NBC Report
- Shell Report
- Situation Report
- Spot Report

4.14.8 File: rpt_value.c

Path: /simnet/ivis.ivis-rpt/rpt_value.c

This file defines the sets of report values as follows:

- add/drop shift amounts
- enemy activity types
- activity levels
- commander's intentions
- left/right shift amounts
- numeric values
- observation times
- enemy types
- ammunition types
- friend types
- obstacle types
- NBC attack types
- NBC burst altitudes
- flast-to-bang times
- NBC shell quantities
- nuclear crater sizes
- nuclear cloud widths

4.15 Ivis-Fil Directory

Path: /simnet/ivis/ivis-fil

4.15 1 File: fil_main.c

Path: /simnet/ivis/ivis-fil/fil_main.c

This file contains the code that creates, selects, reads, saves, writes, and updates a report file.

FUNCTION NAME	DESCRIPTION
CreateIvisFile	given a file description, open the file, create a list widget, if needed, and load the data structure
SelectFileEntry	gets the number of the entry selected from the list and reads the entry from the file
ReadEntryFromFile	get an entry from the route/report file
SaveMessage	save a route/report that was received
WriteEntryToFile	given data, output it to a file
AddListString	given a text string, add it to the specified position
AddFileAndListPosition	make a new entry in the FileAndListPosition list
DeleteEntryFromFile	remove entry from a list and file, the entry is removed from the file by writing the last entry in the file to the deleted entry position
RecoverIvisFile	down time are added to route/reports in files, FileAndListPosition list is recreated, and scrolling is recreated

4.15.2 File: fil_rpt.c

Path: /simnet/ivis/ivis-fil/fil_rpt.c

This is the code which creates and maintains old report files and menus

FUNCTION NAME	DESCRIPTION
CreateOldReportMenu	creates menu for looking at old report files
CreateIvisReportFiles	for each report type, create a file to save created and received reports
ShowReportFileMenu	shows the report file menu with the correct scrolling list and title
ShowReportIcons	a report was selected, show its icons
HideFileMenu	unmap the list widget and hide the menu
SaveCreatedReport	given report type and report, save it
SaveReceivedReport	save report from message queue with other data from the message queue
ShowOldReport	call ShowReport to display the selected report
DeleteOldReport	remove selected report from file and list
RecoverReportFiles	load the files into the scrolling lists

4.16 Ivis-Map Directory

Path: /simnet/ivis/ivis-map

4.16.1 File: map.init.c

This file contains routine for processing resource database entries to obtain information about how the map display is configured. It includes IVIS standard features grouping, colors for standard features, and map characteristics resources.

FUNCTION NAME	DESCRIPTION
InitMapDisplay	initializes the standard features of the map display for Ivis
Cvt_string_to_interval	this routine takes a string consisting of integer pairs and converts it to an array of map scale/interval structure which can then be used by the gridline and contour modules for determining intervals between gridlines and contour lines
Cvt_string_to_integers	this routine takes a string consisting of integer pairs and converts it to an array of integers which can then be used by bit module for determining map scales at which there will be preprocessed image files
Cvt_string_to_mapftr_bits	This routine takes a string consisting of standard feature names and returns a MAPFTR_BITS bit vector with the appropriate bits turned on

4.16.2 File: map_main.c

Path: /simnet/ivis/ivsi-map/map_main.c

This file contains routines for processing user interface functions with the map.

FUNCTION NAME	DESCRIPTION
CreateMapPane	the map is created in a drawing area widget
InitMap	initialization of map window structure
InitMapScrolling	initialization of variables used to draw the scrolling indicator
ChangeMapScale	callback on change of map scale radio buttons
ChangeScrollMode	callback on change of map scroll radio buttons
CHangeFeatureVisibility	callback on change of map feature toggle buttons
SetFeatureVisibility	change the specified visibility
MapChanged	notify ivis of the need to redraw the map window
IsChangeMap	returns the reason for redrawing the map; 0 if the map does not need to be redrawn
NotifyMapChange	refreshes, rescales, and repositions the map
MapExposeHandler	handle an expose event in the map window
MapInputEvent	handle a mouse button event on the map pane
RedrawMap	called to redraw map
Constrain_map_ogrin	
MapMenuVisibility	
MapCenter	
MapRepositionEnabled	
ActiveMapReposition	
EnterActiveReposition	
MapReposition	called to display reposition map message
MapScrolling	called to display scrolling map message
MapAttachPoint	called to display attach message
LeaveActiveReposition	
UpdatAttachOffset	
UpdateAttachLocation	
UpdateMapLocation	
SendMapStats	get the state of the map (scale, scroll mode, and features), and send it away to be collected
CheckpointMap	save map parameters describing the state of the map and map statistics
RecoverMap	recover map parameters and statistics

4.16.3 map_menu.c

Path: /simnet/ivis/ivis-map/map_menu.c

These are the routines that define map menus. It defines the scale of the buttons, the value of the scale group, the value of the scroll buttons, the value of the baseline scroll buttons, defines the mapfields, and the contents of the map features menu.

4.17 Ivis-Nav Directory

Path: /simnet/ivis/ivis-nav

4.17.1 File: nav_draw.c

Path: /simnet/ivis/ivis-nav/nav_draw.c

This file contains the code for drawing and updating the active route (solid line). It also provides drawing proposed routes (dashed line - routes from the message queue and routes from the route file).

FUNCTION NAME	DESCRIPTION
SetProposedRoute	update proposeRoute to selected route, receive route, or none, if the route file menu is exposed, then "selectedRoute" is the proposed route; if a route in the received queue is being shown, then "receiveRoute" is the proposed route; otherwise there is no proposed route
DrawAllNavRoutes	called from the RedrawMapOver function to redraw navigation routes
DrawNavRoute	draw a route, draw an icon for each waypoint, and a line between successive waypoints
DrawDriveToLinw	draw the line from the vehicle to active waypoint
RedrawActiveRte	callback when a waypoint in the active route has changed
AlterActiveRte	given a waypoint number, draw the waypoint label and the connecting lines

4.17.2 File: nav_drive.c

Path: /simnet/ivis/ivis-nav/nav_drive.c

This file contains the code which communicates to the driver's display.

FUNCTION NAME	DESCRIPTION
StartDriverDisplayUpdate	initialize the packet for the driver's display
UpdateDriverDisplay	update the driver's display

4.17.3 File: nav_file.c

Path: /simnet/ivis/ivis-nav/nav_file.c

This module contains the code which saves, recalls, and deletes routes from the route file.

FUNCTION NAME	DESCRIPTION
SelectRoute	erase the old selected route, get the new one and draw it
DisplaySelectRoute	load the route tbd coordinates into the menu
SaveReceiveRoute	save a route forced from the message queue
Save RouteMessage	save a received route which user decided to save
SaveCreatedRoute	save the route in the route file
DeleteSavedRoute	erase the route from the map and remove it from the file

4.17.4 File: nav_main.c

Path: /simnet/ivis/ivis-nav/nav_main.c

This file contains code describing creation of the navigation menus, initialization or routes, checkpoint and recovery, and miscellaneous functions.

FUNCTION NAME	DESCRIPTION
CreateNavMenus	create the navigation (active), working and routefile menus, initialize the routes and route file, create the action menu, get waypoint bitmaps
InitRoute	Null out the route and set all locations to locationUnknown
ClearField	clear the highlighted waypoint field
ActivateRoute	take the given route and make it the active route
MakeRouteQueueString	given route with originator and creation time, make the string for the scrolled list
CheckpointRoute	checkpoint the active route
RecoverRoute	recover the active route

4.17.5 File: nav_menu.c


Path: /simnet/ivis/ivis-nav/nav_menu.c

This file contains the code for describing navigation menus. It also contains the contents of the routes file menu, defines the value for waypoints, describes the waypoint button group, contains the contents of the navigation menu, and contains the contents of the receive route menu.

4.17.6 File: nav_msg.c

Path: /simnet/ivis/ivis-nav/nav_menu.c

This module contains the code which sends and receives route messages.

FUNCTION NAME	DESCRIPTION
SendRoute	send the route (callback parameter) to another unit
SendRoutePacket	send the route packet after send button is pressed in send menu
ShowRouteMessage	display the route chosen in the Receive queue
CancelShowMessage	exit  which displays a route message
DeleteShowMessage	delete a route message from the receive queue
CancelRoutePacket	cancel sending a route (cancel in action menu)

4.18 Bitmaps Directory

Path: /simnet/ivis/bitmaps

This directory contains the bitmaps for the target icons and waypoints. Each file contains only one bitmap and the files are as follows:

File Name (Bitmap)	DESCRIPTION
arty	icon for enemy artillery
atgm	icon for enemy ant-tank guided missiles
biocloud	icon for biological cloud
chemcloud	icon for chemical cloud
curloc	splat for highlighting a location
friend	
fwair	icon for enemy fixed-wing aircraft
helo	icon for helicopter
nucloud	icon for nuclear cloud
obsloc	
pc	icon for enemy personnel carrier
shell	
tank	
target	icon for enemy target of unknown type
troop	icon for enemy troops
truck	icon for enemy truck
wp1	
wp2	
wp3	
wp4	
wp5	
wp6	

4.19 Overlays Directory

4.20 Resources Directory

Path: /simnet/ivis/resources

This directory contains the files for ivis resources.

File Name (Bitmap)	DESCRIPTION
action	report sending menu
color.full	this file defines resource values for operating the IVIS simulation in full color mode
color.mono	this file defines resources values for operating the IVIS simulation in monochrome (amber on black) mode
config.sim	this file defines the configuration options which are specific to a particular IVIS simulation
config.std	this file defines the standard configuration options which are common across all simulators
display	resources defining the overall appearance
filelist	
map	colors of map features are defined in monochrome or color
map.full	this file defines resources values for operating the IVIS simulation
map.grid	this file defines resource values for operating the IVIS simulation in grid-only map mode
map.base	these resources tailor IVIS for operation in baseline mode
mode.dev1	these resources tailor IVIS for operation in development mode
mode.enhc	these resources tailor for operation in enhanced baseline mode
mode.expr	these resources tailor IVIS for operation in experimental mode
navigation	this file contains the resources for the navigation menu, route files menu, and received message menu
receive	this file contains the amount of time (in minutes) messages sit in the queue before automatically being placed in files
report	this file contains the resources for report menu and report file menu

rpt.new	this file contains the resources for adjust fire report creation menu, ammunition status report menu, fire report creation menu, contact report creation menu, Frago report creation menu, intelligence report creation menu, shell report creation menu, situation report creation menu, spot report creation menu, and NBC report
rpt.show	this file contains the resources for adjust fire report viewing menu, ammunition status report creation menu, call for fire viewing menu, contact report viewing menu, FRAGO report viewing menu, intelligence report viewing menu, shell report viewing menu, situation report viewing menu, spot report viewing menu, and NBC show menus
status	resources defining the appearance of items in the status pane
widget	resources defining the behavior of standard widgets

4.21 Scripts Directory

Path: /simnet/ivis/scripts

This directory contains the files of scripts for distributing, back-up, calibrating, and starting ivis.

File Name	DESCRIPTION
dist-ivis	this script distributes Ivis files to specified hosts.
make-backup	this script makes a back-up of the Ivis system
make-dist	
start-cal	this script runs the touch-screen calibrator
start-ivis	this script starts the Ivis simulator

APPENDIX A: DIRECTORY, HEADER, and LIBRARY NAMES

DIRECTORY NAME	HEADER FILE NAME	LIBRARY NAME
ivis-com	com_ivis.h	libcom.a
ivis-fil	fil_ivis.h	libfil.a
ivis-inp	inp_ivis.h	libinp.a
ivis-main	ivis.h	none
ivis-map	map_ivis.h	libivismap.a
ivis-menu	menu_ivis.h	libmenu.a
ivis-misc	misc_ivis.h	libmisc.a
ivis-nav	nav_ivis.h	libnav.a
ivis-ovr	ovr_ivis.h	libovr.a
over-icn	none	libicn.a
over-cm	none	libcm.a
cm-lines	none	liblines.a
cm-symbols	none	libsymbols.a
ivis-pane	pane_ivis.h	libpane.a
ivis-rpt	rpt_ivis.h	librpt.a
ivis-util	none	none
point	point.h	libpoint.a
bitmaps	none	none
overlays	none	none
resources	none	none
scripts	none	none

APPENDIX B: IVIS SOFTWARE COMPONENT CONFIGURATION

